

일반논문-06-11-1-06

디지털 방송 TV수신기의 기능 업그레이드를 위한 소프트웨어 다운로드와 설치 기능 구현

류일권^{a)†}, 정문열^{a)}, 김정환^{a)}, 최진수^{b)}, 방건^{b)}

Implementation of Software Downloading and Installing for upgrading Digital TV Settop Box

Yil Kwon Ryu^{a)†}, Moon Ryul Jung^{a)}, Jung Hwan Kim^{a)}, Jin Su Choi^{b)}, and Gun Bang^{b)}

요 약

디지털 방송과 데이터 방송 기술이 지속적으로 발전함에 따라, 디지털 방송에 신기술이 도입되고 새롭게 시도되는 방송 서비스가 계속 등장할 것이다. 이미 가정에 출시된 STB가 새롭게 도입되는 신기술과 서비스를 계속 수용할 수 있으려면, 이때마다 기존에 탑재된 STB의 소프트웨어는 새로 업그레이드되어야 한다. 일반적으로 STB가 한번 가정 내에 보급되면 STB에 탑재된 기존의 소프트웨어는 업그레이드되기가 용이하지 않다. 본 논문은 방송망의 전송 채널을 통해 STB의 소프트웨어를 다운로드 받아 이를 업그레이드 시키는 소프트웨어 업데이트 시스템을 제시하고 구현함으로써, 현재 용이하지 않는 STB의 소프트웨어 업그레이드 환경을 극복한다. 제안하는 소프트웨어 업데이트 시스템은 데이터 카루셀 스트림에서 업데이트될 소프트웨어를 다운로드 받는 (1) 다운로드(Downloader), 다운로드된 소프트웨어를 설치하는 (2) 업데이트 로더(Update Loader), 그리고 예외상황이 발생하면 STB가 새로 부팅될 때, 로그 파일을 이용하여 소프트웨어를 이전 상태로 복구시키는 (3) 리커버러(Recoverer)의 세 가지 모듈로 구성된다. 소프트웨어 업데이트 시스템에서 다운로드되는 지상파 디지털 방송 규격인 ATSC에서 제시한 소프트웨어 다운로드 데이터 서비스 규약(A/97)에 맞게 구현하고 ATSC 환경에서 이를 실험 검증하였다.

Abstract

As constant development of digital broadcasting and data broadcasting system, new technology will be introduced to digital broadcasting and new broadcasting service will be appeared. These services need to be changed and processed to suit each services and the software of the receiver need to be upgraded. Though, generally the digital broadcasting receivers are not getting updated once it's delivered to home for long time and it need to be done by hand or collected each one of them with trouble. Therefore this paper suggests a way to overcome these difficulties via broadcasting stream. This research is to describe how three modules - namely (1) Downloader, downloads new software from data carousel stream, (2) Update Loader, installs the software received by downloader; and (3) Recoverer, recovers the former version of the software if some serious problem has been occurred during downloading and installing the software. This paper tries to realize the accommodation of terrestrial STB based on the new technique and service following ATSC A-97 agreement.

Keyword: Software Download Data Service, ATSC, A/97

a) 서강대학교 영상대학원 미디어공학

Media Lab Dept of Media Technology, Graduate School of Media Communications Sogang University

b) 한국전자통신연구원

Electronics and Telecommunications Research Institute

† 교신저자 : 류일권(acesryu@naver.com)

I. 서론

디지털 방송 서비스는 지상파, 위성, 케이블 등의 방송 매체나 지역에 따라 서로 다른 규격들을 기반으로 제공되고 있다. 현재 우리나라의 경우 지상파 방송은 ATSC (Advanced Television Systems Committee), 케이블 방송은 OpenCable, 그리고 위성 방송은 DVB(Digital Video Broadcasting)를 각각의 디지털방송 표준으로 정하고 있다. 또한, 고화질, 고음질의 비디오/오디오 프로그램과 함께 멀티미디어 데이터를 처리할 수 있는 수신장치 미들웨어 규격으로 DVB-MHP(Multimedia Home Platform), ATSC-ACAP(Advanced Common Application Platform), OCAP(OpenCable Application Platform)을 각각 데이터 방송 표준으로 확정 혹은 잠정적으로 정하고 이를 위한 STB가 출시되어 사용되고 있다. 이러한 규격을 기반으로 방송매체를 통한 전자상거래, VOD, 채팅 등 다양한 서비스들이 가능하게 되었다. 이와 같은 다양하고 새로운 서비스가 등장했을 때마다 이를 수용하기 위해 소프트웨어 업데이트가 유연한 STB가 필요하다. 그러나 일반적으로 디지털 방송 수신 장치는 한번 각 가정으로 보급되면 오랜 기간 동안 교체하지 않고 사용되며 STB의 기능 추가 및 성능 향상을 위한 업그레이드 작업을 위해서는 직접 수작업을 통해 하거나 STB를 수거해야 하는 등 많은 어려움이 따르게 된다. 또한, 이로 인해 새로운 서비스 도입 시기가 늦어지게 되어 사업성이 떨어질 뿐만 아니라 사용자에게도 시간적, 경제적 부담을 주게 된다.

이에 본 논문은 새로운 기술이나 서비스에 쉽게 적용될 수 있도록 STB 내에 상주하는 middleware, native application software 등의 업그레이드를 위한 프로그램, 다시 말해 방송 스트림을 통한 STB의 업데이트를 실현시키는 소프트웨어 업데이트 시스템을 ATSC 기반으로 구현하고자 한다.

본 논문은 2장에서 소프트웨어 업데이트를 위한 데이터 카루셀 스트림 생성 방법과 이와 관련된 ATSC 규약에 대해 설명을 한다. 3장에서는 소프트웨어 업데이트 시스템의 구현 방법을 설명하고 DVB에서 구현되고 있는 소프트웨어 업데이트와의 차이점을 비교한다. 마지막으로 4장에서는 실험 환경 및 결과에 대해 서술하였다.

II. 소프트웨어 다운로드 데이터 서비스

소프트웨어 다운로드 데이터 서비스는 ATSC - A/97 의 동명 문서에서 규정한 방식으로 STB의 펌웨어 (firmware), 운영체제 소프트웨어, 디바이스 드라이버 소프트웨어, 상주 애플리케이션 소프트웨어(native application software), 미들웨어(middleware) 및 기타 소프트웨어 프로그램을 업데이트하기 위한 데이터 서비스의 전송을 말한다^[5]. 국내 지상파 디지털 방송 규약인 ATSC에서 정의한 소프트웨어 다운로드 데이터 서비스는 DSI, DII, DDB 메시지를 이용한 2-layer Data Carousel Scenario를 기반으로 이루어진다^[6]. 본 장에서는 ATSC 방식으로 STB의 소프트웨어를 방송망을 통해 업그레이드시키기 위한 방송 규약에 대해 기술한다.

1. Program and System Information Protocol

디지털 방송에서 서비스는 기존의 아날로그 방송과 달리 비디오/오디오 스트림 이외에 데이터 스트림으로 구성될 수 있다. 이와 같이 여러 개의 스트림이 하나의 전송 스트림내에 다중화되어 있으므로 STB가 여러 스트림에서 특정 스트림을 구분해 내기 위해서 다중화되어 있는 전송 스트림의 구조 정보를 제공 받아야 한다. PSIP(Program and Systems Information Protocol) 테이블들은 디지털 방송 채널 이름, 채널 번호, 채널 이동 등의 정보를 제공하기 위한 ATSC 표준으로써 전송 스트림의 구조 정보를 제공한다. PSIP은 STT(System Time Table), RRT(Rating Region Table), MGT(Master Guide Table), VCT(Virtual Channel Table)의 기본 테이블들로 구성되며 같은 PID(0x1FFB)값을 가진다^[2]. PSIP 테이블들은 MPEG-2 시스템의 "Private Section"의 형식을 따른다^{[8][10][12]}.

그림 1 은 각 PSIP 테이블에 대한 상호간의 계층도를 나타내는 것으로써 STB가 각 PSIP테이블에 접근하는 방법을 제시한다. 그림 1 에서 base PID은 STT, RRT, MGT, VCT 테이블을 전송하는 전송 스트림의 PID로써 명시적으로 0x1FFB값을 가진다. STT, RRT, MGT, VCT가 동일한 PID를 가지기 때문에 STB는 Table ID로 이 테이블들을

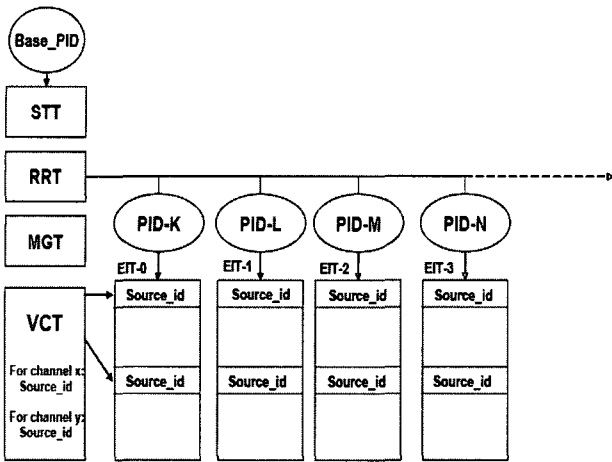


그림 1. PISP 테이블들에 대한 상호간의 계층도
 그림 1. Hierarchy for the Program and System Information Protocol^[2]

구분한다^{[10][12]}. EIT나 ETT의 경우 MGT 내에서 각각 서로 다른 PID값으로 정해져 있다.

소프트웨어 다운로드 데이터 서비스의 중요 정보가 제공되는 PSIP 테이블은 VCT이다. VCT는 전송 스트림내에 모든 가상 채널에 대한 정보들을 나열한다. VCT는 물리적인 채널을 통해 전달되는 여러 개의 가상 채널에 대한 정보를 포함한다. 즉 가상채널의 이름과 채널에 구성되어 있는 방송 프로그램의 종류, 채널전달 변조방식, 주파수 및 개별 방송 프로그램에 해당하는 source_id 정보를 전달한다. 부가적인 정보는 기본적인 정보가 전달된 후에 디스크립터 루프 형태로 전송된다. VCT의 디스크립터 루프에서 정의되는SLD(Service Location Descriptor)는 각 가상 채널을 구성하는 성분 스트림들에 대한 정보, 즉, 스트림의 종류, PID등을 정의한다^[2].

2. 데이터 카루셀 프로토콜

ATSC는 소프트웨어 다운로드 데이터 서비스를 전송하기 위한 프로토콜로 데이터 카루셀을 채택하고 있다. 데이터 카루셀은 한정된 양의 카루셀 내 데이터들을 순환적으로 반복해서 전송해 주는 방식이다^[9]. 데이터 카루셀은 통지 메시지(Control Message)와 다운로드 데이터 메시지(Download Data Message)로 구성된다^[7]. 통지 메시지는

데이터 카루셀의 구조 정보를 제공하여 수신기가 데이터를 수신할 수 있도록 해주는 메시지이고, 다운로드 데이터 메시지는 실제 데이터가 담겨 있는 메시지이다^{[3][4]}.

데이터 카루셀은 모듈 단위로 전송한다. 데이터 카루셀은 이 모듈들을 전송하기 위하여 DDB 메시지 단위로 모듈을 분리하게 된다. STB의 모듈 수신을 위해 각각의 모듈들에 대한 정보를 수신기에 제공해주는 통지 메시지는 DSI(Download Server Initiate), DII(Download Info Indication) 메시지가 사용된다^[7]. 데이터 카루셀의 구조는 전송할 모듈의 특징에 따라 통지 메시지와 다운로드 데이터 메시지를 통해 1혹은 2개의 계층으로 나누어 구성할 수 있다. 데이터 카루셀에서 상위 메시지는 자기에 속한 하위 메시지에 대한 식별값을 정의하여 각 계층에 대한 그룹화를 수행한다. DII 메시지는 각각의 모듈들과 DDB 메시지에 대한 정보를 제공한다. 데이터 카루셀이 전송할 모듈의 수가 많아 하나의 그룹에 들어갈 수 없거나 관리상의 편의를 제공하기 위해 데이터 카루셀을 여러 그룹으로 나누어

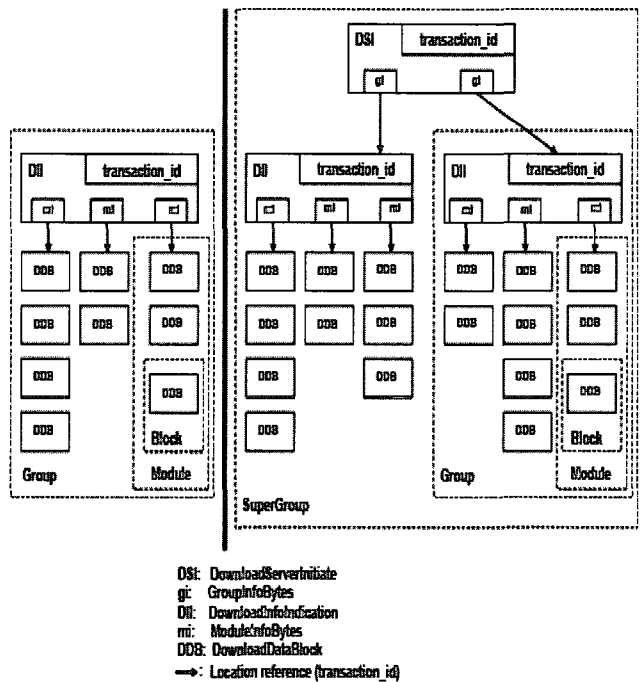


그림 2. 1계층과 2계층 다운로드 시나리오의 메시지 구조
 Fig. 2. Structure of the One-Layer and Two-Layers Control Information Download Scenario^{[3][4]}

구성할 수도 있다. 이 때 각각의 그룹에 대한 정보를 제공하기 위해 DSI 메시지가 데이터 카루셀에 포함된다^{[3][4]}.

소프트웨어 다운로드 데이터 서비스는 2계층 데이터 카루셀(Two-Layers Data Carousel)로 전송이 되는데 2계층 데이터 카루셀은 DSI 메시지, DII 메시지, DDB 메시지 들로 구성이 된다. DSI 메시지는 여러 하위 그룹들을 설명하는데 사용된다. DSI 메시지는 GroupInfoIndication 구조에서 각 그룹들의 여러 정보를 제공하며 보조적인 정보는 groupInfoByte에 Descriptor들의 리스트를 통해 제공된다. DII 메시지는 ModuleInfoByte 필드를 사용하여 다운로드 시나리오의 모듈들을 기술한다. 그림2에서 화살표는 화살표가 가리키는 메시지를 얻어오기 위한 접근 정보를 표현한다. 수신기는 스트림으로 부터 메시지를 효과적으로 얻어 내기 위해 이 값들을 사용할 수 있다. 상위 그룹내의 모든 DDB와 DII 메시지는 모두 동일한 downloadId 가지며, 모든 모듈의 moduleId는 같은 downloadId에서 유일한 값을 가진다. 각각의 통지 메시지는 메시지의 유일한 id인 transaction id값을 가진다. 수신기는 transaction id와 moduleId를 이용해 데이터 카루셀에서 실제 모듈 데이터를 얻어 낼 수 있다. 각 모듈은 하나 이상의 DDB 메시지의 페이로드 나누어 담겨 전송이 된다. DDB 메시지가 담을 수 있는 모듈 데이터의 최대 크기는 4 바이트 정도이기 때문에 그 이상의 크기를 가진 모듈은 여러 DDB 메시지에 나누어져 전송이 된다. 이 때 나누어진 DDB 메시지는 모듈의 마지막 부분을 담고 있는 DDB 메시지를 제외하곤 모두 크기가 동일하다^{[3][4]}.

3. 소프트웨어 다운로드 데이터 서비스의 전송

방송 전송 채널에서 소프트웨어 다운로드 데이터 서비스의 전송은 Announcement, Signaling, Encapsulation 과정을 통해 설명될 수 있다^[5]. 소프트웨어 다운로드 데이터 서비스의 Announcement, Signaling, Encapsulation는 데이터 카루셀의 통지 메시지(Control Message)와 다운로드 데이터 메시지(Download Data Message)를 이용하여 수행된다. Announcement와 Signaling은 통지 메시지로 수행이 되고 Encapsulation는 다운로드 데이터 메시지로 수행이 된다^[5].

Announcement는 현재는 전송되고 있지는 않지만 미래에 있을 소프트웨어 다운로드를 미리 예고하는 것으로써, 소프트웨어 다운로드 데이터 서비스가 실시될 미래의 시점에 대한 스케줄 정보를 Schedule Descriptor를 이용하여 수신기에 제공한다^[5]. 데이터 카루셀을 구성하는 그룹들에 대한 스케줄 정보의 통지는 DSI 메시지의 디스크립터 루프에 Schedule Descriptor를 통해 수행된다. 한 그룹에 속하는 모듈들의 전송 스케줄에 대한 통지는 DII메시지를 이용하여 수행된다. 하나의 DII 메시지는 다운로드 할 소프트웨어의 그룹과 대응이 되는 것으로써 각 그룹은 하나의 제작사와 대응된다.

Announcement가 미래에 있을 소프트웨어 다운로드를 미리 예고하는 것임에 반해 Signaling은 소프트웨어 다운로드 데이터 서비스가 현재 방송 스트림으로 전송되고 있음을 수신기에 알리는 것이다. 따라서 STB는 Announcement 메시지를 받으면, 소프트웨어 다운로드 데이터 서비스를 처리하는 미들웨어 모듈들과 자원들을 메모리에서 해제하고 Signaling 메시지를 받기 직전에 다시 메모리에 로드 해야 한다. Encapsulation은 소프트웨어 다운로드 데이터 서비스를 전송하는 송신 측에서 실제 다운로드 될 소프트웨어 데이터를 전송 스트림의 페이로드에 바인딩(Binding)하는 과정이다. STB는 전송 스트림의 페이로드에 바인딩 된 데이터를 추출하여 전송 스트림에 Encapsulation된 실제 소프트웨어 데이터를 처리하게 된다.

4. COMPATIBILITY DESCRIPTOR

Compatibility Descriptor (groupCompatibility)는 데이터 서비스에 대한 STB의 소프트웨어/하드웨어 요구사항을 기술한 서술자이다^{[3][4]}. Compatibility Descriptor에서 제공되는 정보는 전송되는 데이터 서비스를 수신하여 처리할 수 있는 STB의 하드웨어/소프트웨어의 모델, 버전, 제조사 고유번호 등이 제공된다. 소프트웨어 다운로드 데이터 서비스에서 Compatibility Descriptor는 Announcement와 Signaling 단계 모두 에서 사용될 수 있으며, 소프트웨어가 다운로드 되어 업데이트되는 수신기의 필요 정보를 기술하는데 활용된다.

Compatibility Descriptor의 구조는 STB의 하드웨어와 소프트웨어 정보들을 개별적으로 제공할 수 있도록 디스크립터 루프 형태로 되어 있다. A/97에서는 소프트웨어 다운로드 데이터 서비스의 descriptorCount 필드는 1이상이어야 한다고 규정한다⁵⁾. 즉 소프트웨어 다운로드 데이터 서비스는 Compatibility Descriptor를 반드시 전송 스트림에 포함시켜야 한다. Compatibility Descriptor는 데이터 카루셀에서는 보통 사용하지 않는다. 모든 STB 하드웨어/소프트웨어 모델에서 일반적인 데이터 카루셀은 처리될 수 있다고 보기 때문이다. 그러나 소프트웨어 다운로드 데이터 서비스는 특정 하드웨어/소프트웨어가 탑재된 STB의 업그레이드를 목적으로 실행되는 서비스이기 때문에, Compatibility Descriptor가 소프트웨어 다운로드 데이터 서비스에서는 데이터 카루셀에 반드시 포함된다.

Compatibility Descriptor에서 제공되는 specifierType, specifierData, model, version은 전송되는 소프트웨어 다운로드 데이터가 어떤 종류의 STB들을 대상으로 하고 있는지에 대한 정보를 제공한다. STB는 Compatibility Descriptor에 기술된 위의 정보가 STB의 그것과 일치할 때만 데이터를 다운로드 받는다. 위 형식의 Compatibility Descriptor는 실제로는 DSI 메시지의 groupInfoIndication 구조 안에서 기술되어 각 그룹의 특징을 기술하는데 쓰인다. 이렇게 쓰이는 Compatibility를 groupCompatibility 라 한다³⁾⁴⁾.

5. 소프트웨어 다운로드 데이터 서비스를 위한 VCT와 데이터 카루셀 구성

소프트웨어 다운로드 카루셀 스트림이 전송되는 가상 채널의 service_type은 VCT에서 0x05로 설정된다. service_type이 0x05인 가상 채널의 SLD(Service Location Descriptor)에서 소프트웨어 다운로드 카루셀 스트림의 PID가 지정된다⁵⁾. 소프트웨어 다운로드 데이터 카루셀의 DSI, DDI, DDB 메시지가 모두 이 PID로 전송이 된다. 소프트웨어 다운로드 데이터 서비스는 2계층 데이터 카루셀 구조로 전송이 되므로 DII 메시지가 하나라도 DSI 메시지

는 반드시 필요하다. DSI 메시지에서 각 그룹에 대한 가장 중요한 정보는 groupCompatibility 와 groupInfoByte 에서 제공된다. groupInfoByte는 각 그룹의 특징을 기술하는 디스크립터들의 리스트로 구성된다. Announcement DSI메시지인 경우에는 scheduleDescriptor가 groupInfoByte 내에서 정의될 것이다. scheduleDescriptor는 해당 그룹이 언제 전송되기 시작할 것이며, 전송을 완성하는데 얼마나 오랜 시간이 걸릴 것인가를 기술한다. 이런 DSI 메시지는 최소 60초 마다 한번씩 전송되도록 한다. 현재 진행중인 다운로드에 대한 정보를 제공하는 Signalling DSI 메시지는 groupInfoByte 필드를 구성하는 Descriptor 리스트에 Announcement DSI의 경우와 달리 scheduleDescriptor를 제공하지 않는다.

소프트웨어 다운로드 서비스의 카루셀 스트림에서 하나의 DII 메시지는 하나의 제작사와 대응된다. DII 메시지와 대응되는 제작사의 정보는 DSI 메시지의 groupCompatibility 에서 제공된다. DII 메시지의 moduleInfoByte는 모듈들에 대한 정보를 제공하는 Descriptor 리스트로 구성이 된다. DDB 메시지는 실제 업데이트 되는 소프트웨어를 전송하기 위해 각 모듈을 구성하는 블록들을 메시지로 담고 있다. 소프트웨어 다운로드 데이터 서비스에만 적용되는 특별한 DDB 메시지의 제약사항은 없고 다른 일반 데이터 카루셀 스트림에 대한 적용을 그대로 따른다.

Ⅲ. STB의 소프트웨어 업데이트 시스템

본 논문에서 제안하는 소프트웨어 업데이트 시스템은 ATSC A/97 규약에 따라 인코딩된 다운로드 카루셀 스트림으로부터 소프트웨어를 단계적으로 추출하는 (1)다운로더, 다운로드된 소프트웨어를 STB에 설치하는 (2)업데이트 로더, 업데이트 로더가 STB의 소프트웨어를 설치하는 과정에서 예외상황이 발생하여 소프트웨어 업데이트가 실패되었을 때, STB의 소프트웨어를 이전 상태로 복구시키는 (3)리커버러 등으로 구성된다. 본 장에서는 소프트웨어 업데이트 시스템의 각 모듈의 구현에 대해 기술한다.

1. SMARTV의 섹션 필터링

다운로더는 TeraLogic의 Cougar 라이브러리를 이용하여 ATSC 소프트웨어 다운로드 데이터 서비스 규약인 A 97를 수신기에서 구현하였다. TeraLogic은 본 논문의 구현 플랫폼인 SMARTV STB의 DTV 수신칩셋 제조사이다. TeraLogic의 Cougar 라이브러리는 다운로더가 구현되어 탑재되는 SMARTV 셋탑 박스의 소프트웨어 개발 플랫폼으로써 튜너(Tuner), 디모듈레이터(Demodulators), 디멀티플렉서(Demultiplexer) 혹은 디믹스, 섹션 필터(Section Filter), MPEG-2 오디오/비디오 디코더 등의 STB에서 전송 스트림을 처리하는 하드웨어를 소프트웨어적으로 제어할 수 있는 환경을 제공한다.

다운로더는 데이터 카루셀로부터 소프트웨어를 다운로드 하기 위해 단계적으로 VCT, DSI, DII, DDB 등을 Cougar 라이브러리를 이용하여 섹션 필터링한다. 다운로더는 섹션 필터링 과정을 통해 원하는 섹션 데이터를 얻어 내어 각 바이트 배열을 ATSC 규약에 의거하여 처리함으로써 소프트웨어 다운로드를 수행한다. 따라서 다운로더의 중요한 처리 과정은 섹션 필터링으로 시작된다.

본 논문의 구현 환경인 SMARTV 셋탑 박스에서 섹션 필터링이 수행되려면 디믹스가 초기화되어야 한다. 디믹스 초기화는 SMARTV 셋탑 박스의 디믹스가 전송 스트림을 입력받을 수 있도록 준비를 하는 것이다. 디믹스의 초기화는 셋탑 박스가 부팅이 될 때 다운로더의 구동보다 우선적으로 처리되어야 할 것이다. 본 논문에서 디믹스 초기화는 독립적인 프로세스에서 Cougar 라이브러리를 이용하여 실행되었고 디믹스가 초기화된 상태에서 다운로더가 실행되도록 구현하였다. 디믹스가 초기화된 이후, 다운로더는 섹션 필터에 필터링할 섹션의 PID와 Table ID 값을 인자로 넘겨 주어 원하는 섹션을 필터링하게 된다.

TeraLogic의 Cougar 플랫폼에서 섹션 필터링 수행은 3 단계의 필터 작업을 거친다. 3단계의 필터 작업은 PID 필터, 섹션 필터, 테이블 필터들에 의해 수행된다. 우선 요청할 섹션의 PID를 섹션 필터에게 인자로 넘겨준다. 섹션 필터는 인자로 넘겨 받은 PID를 디믹스의 PID 필터에게 다시 인자로 건네 주게 되고, PID 필터는 해당 PID의 TS

Packet만을 섹션 필터 쪽으로 넘겨주게 된다. 섹션 필터는 디믹스의 PID 필터로부터 건네 받은 TS Packet 을 취합하여 섹션 데이터를 만들어 섹션 필터링을 요청한 쪽에 바이트 배열 형태로 넘겨 준다. 2개 이상의 섹션으로 구성된 테이블은 테이블 필터에 의해 만들어 진다. 테이블 필터는 섹션 필터에 의해 추출된 섹션 데이터를 재조합하여 테이블 형태로 만들어 준다. 섹션 필터가 다운로더에게 섹션 데이터를 넘겨주는 과정은 비동기적인 처리에 의해 이루어진다. 비동기적인 처리는 다운로더가 섹션 필터링을 요청할 때 등록된 이벤트 핸들러가 호출됨으로써 이루어진다. 따라서 다운로더는 등록된 이벤트 핸들러에 섹션 데이터를 처리하도록 구현하였다.

2. TL850 MPEG Transport Driver의 제어와 관리

TL850은 SMARTV STB의 내장형 Transport Stream Demultiplexer(TSD)로써 마이크로 컨트롤러 형태로 구성된 ASIC 칩에 내장되어 있다. TL850은 전송 매체에 상관없이 다중화된 전송 스트림에서 오디오, 비디오, 데이터, SI(System Information) 정보 등을 각 성분 스트림으로 역다중화하여 STB내 성분 스트림 처리 모듈(A/V 디코더, 섹션 필터)에게 각 성분 스트림을 전달한다. 다운로더의 주요한 임무는 소프트웨어 다운로드과정에서 필요한 섹션들을 필터링하고 전송 스트림을 처리하는 것이다. 섹션 필터링할 때, TL850은 섹션 필터와 더불어 가장 중요한 역할을 하는 하드웨어 자원이므로 다운로더 구현에서 TL850과 섹션 필터에 대한 적절한 제어가 가장 중요하다. 다운로더의 TL850과 섹션 필터 하드웨어 자원에 대한 잘못된 제어와 관리는 원활한 섹션 필터링과 전송 스트림 처리를 저해한다. 뿐만 아니라 TL850과 섹션 필터 자원들은 STB의 다른 모듈에서도 공유해서 쓰는 자원이므로 전체 STB의 오동작을 불러일으킬 수 있다^{[11][13]}.

TL850과 섹션 필터 하드웨어는 STB의 다운로더 이외의 다른 모듈에서도 공유하여 쓰기 때문에 별도로 이 자원들만 관리하는 모듈을 따로 구현함으로써 다른 모듈들의 TL850과 섹션 필터 하드웨어 자원에 대한 직접적인 접근을 제한하였다. TL850과 섹션 필터를 관리하는 이 모듈은

다른 모듈의 섹션 필터링 요청을 대신하여 수행하고 섹션 필터링 결과와 섹션 데이터를 요청한 쪽에 프로세스 간 통신을 통해 전달한다.

다운로더가 DSMCC 섹션을 필터링하는데 이용하는 Cougar 라이브러리 함수는 MonitorPSIPid이다. 이 함수 원형에는 여러 함수 인자가 정의 되어 있는데 데이터 카루셀을 필터링하는데 가장 중요한 인자는 DSMCC 데이터 카루셀 스트림의 PID, 비트 필터 마스크 조건, 이벤트 핸들러, 여러 섹션 필터 요청에 부여되는 고유 ID 등이다. DSMCC 데이터 카루셀 스트림의 PID는 실제 데이터 카루셀이 전송되는 전송 스트림의 물리적 위치값인 PID를 설정한다. 비트 필터 마스크 조건은 DSI와 DII처럼 PID로 구별할 수 없는 서로 다른 섹션들을 필터링하기 위해 좀 더 자세한 필터 조건을 설정한다. 비트 필터 마스크 조건에 많이 이용되는 것은 섹션의 Table ID이다. 데이터 방송에서 섹션 필터링 처리는 비동기적으로 이루어진다. 섹션 필터링이 진행되는 도중 섹션 필터링 조건에 부합되는 섹션이 검출되면 이를 섹션을 요청한 곳에 알려야 한다. 비동기적 처리는 MonitorPSIPid에 이벤트 핸들러 인자로 넘겨준 함수 포인터를 호출함으로써 이루어진다. 섹션 필터 요청에 부여되는 고유 ID는 섹션 필터 작업을 취소 시킬 때 이용되는 ID이다. 원하는 섹션 데이터를 얻어 필요한 정보를 얻은 다음에는 반드시 CancelPSIPidMonitor를 호출하여 요청한 섹션 필터 작업을 중단하여야 한다. CancelPSIPidMonitor 함수 호출시, 함수 인자로 섹션 필터 요청할 때 할당된 ID를 넘겨줘 그 ID에 해당하는 섹션 필터 작업을 종료 시킨다. 명시적으로 CancelPSIPidMonitor를 호출하여 요청한 섹션 필터 작업을 종료 시키지 않으면 섹션 필터가 계속해서 그 섹션 필터링 작업을 수행하기 때문에 그 섹션 필터는 다른 섹션 필터 요청을 수행할 수 없다. 섹션 필터는 여러 STB 모듈이 공유하는 자원이므로 섹션 필터 요청과 취소를 정확히 규칙을 지켜 수행해야 한다. STB의 모든 모듈은 기본적으로 다른 섹션 필터 작업을 요청하기 전에 CancelPSIPidMonitor를 호출하여 이 전에 수행하였던 섹션 필터 작업을 종료 시켜야 한다. STB의 예외 상황에 대해서도 STB의 모든 모듈은 모두 명시적으로 지금 수행하고 있는 섹션 필터 작업을 종료시킨 다음, 다시 그 섹션 필터 작업

을 요청해야 한다.

전송 스트림의 STB 내부 이동 경로에 따라 전송 스트림이 TL850 내부로 들어오기 위한 입력 포트가 다르다. 즉 TL850에 대한 전송 스트림의 출력 포트는 같지만 입력 포트가 다르므로 어떤 경로로 전송 스트림을 받을 지에 따라 TL850의 입력 포트를 우선 설정하고 이 입력포트를 TL850의 출력 포트와 연결해주어야 한다. 실제 전송 스트림이 TL850으로 들어오는 전송 경로의 해당 입력 포트를 출력 포트와 연결해주지 않으면 TL850의 출력 포트로 전송 스트림이 출력되지 않아 이후의 작업을 수행할 수 없다. 따라서 섹션 필터를 요청하는 모듈에서는 TL850의 전송 스트림 입출력 포트를 설정하지 않아야 한다. 실제 전송 스트림이 어떤 경로로 디멀티에 입력되는지 섹션 필터를 요청하는 쪽은 모르기 때문에 섹션 필터를 요청하는 쪽에서 임의로 입출력 포트를 설정하면 안된다. 따라서 본 논문의 STB 소프트웨어 업데이트 시스템은 TL 850의 입출력 포트 설정을 실제로 전송 스트림을 TL850로 보내는 모듈에서 설정하도록 구현되었다.

3. 다운로더의 VCT 처리 모듈

다운로더는 소프트웨어 다운로더 데이터 서비스를 전송하는 카루셀 스트림의 PID를 알아내기 위해 우선적으로 VCT를 필터링한다. VCT를 필터링하기 위한 섹션 필터링 조건은 PID 인자에 0x1FFB, Table ID 인자에 0xC8이 설정된다. VCT가 추출되면 다운로더는 VCT 안에 service type이 0x05 인 virtual channel을 찾는다. 이 가상채널은 데이터 서비스를 전송하는 채널이다. SLD에 해당 가상채널을 구성하는 스트림들의 Stream Type과 그에 대응하는 스트림의 PID가 정의되어 있기 때문에 다운로더는 service type이 0x05 인 가상채널에서 SLD(Service Location Descriptor)를 읽어 들인다. 다운로더는 읽어 들인 SLD에서 Stream Type이 0x0B인 스트림의 PID를 알아낸다. 만약에 service type이 0x05인 가상 채널이 없다면 다운로더는 service type이 0x05인 가상 채널을 VCT에서 발견할 때까지 다시 새롭게 VCT의 섹션 필터링을 수행한다. VCT의 소프트웨어 다운로드 데이터 서비스의 시그널에

대한 다운로드의 실행은 2가지 방식을 생각해 볼 수 있다. 첫 번째 방식으로써, STB의 EPG가 VCT에서 시그널되는 소프트웨어 다운로드 데이터 서비스의 정보를 사용자에게 제공하고 EPG를 통해 사용자가 소프트웨어 다운로드 데이터 서비스를 선택하면 EPG에 의해 다운로드가 실행이 되는 것이다. 두번째 방식으로써, 다운로드는 STB 부팅시 독립적으로 실행이 되어 독자적으로 VCT의 소프트웨어 다운로드 데이터 서비스의 시그널을 받는 것이다. 소프트웨어 다운로드 데이터 서비스가 실행되지 않는 기간에 불필요하게 다운로드가 섹션 필터 자원을 점유하지 않는다는 점에서 첫 번째 방식이 두 번째 방식보다 자원 관리 측면에 있어서 유리하다. 하지만 사용자가 소프트웨어 다운로드 데이터 서비스를 EPG를 통해 선택하지 않는 경우에 대한 STB의 소프트웨어 업그레이드는 해결되지 않는다. 수많은 채널 정보가 제공되는 상황에서 사용자가 소프트웨어 다운로드 데이터 서비스 정보를 정확히 인식할 수 없는 위험이 존재하고 있기 때문에 EPG를 통한 다운로드의 실행은 보급된 전체 STB의 업그레이드를 충분히 보장해 주질 못한다. 반면 다운로드가 독자적으로 소프트웨어 다운로드 데이터 서비스를 처리하는 방식은 섹션 필터 자원 관리 면에서는 불리하지만 STB의 업그레이드가 수신단에서 자동으로 처리되기 때문에 특별한 예외 상황이 아니고서는 (예를 들어 STB의 전원 코드가 소프트웨어 다운로드 데이터 서비스 기간 동안 언플러그드된 상황) 모든 STB의 업그레이드가 보장이 되는 장점이 있다. 다운로드를 구현할 때, 두가지 방식에 따라 특별히 다르게 구현되는 부분은 없다. 단지 두번째 방식의 경우, STB가 동작되는 동안에 다운로드가 독립적으로 계속 동작될 수 있도록 구현되어야 한다.

본 논문의 다운로드의 소프트웨어 다운로드 데이터 서비스를 처리하는 동안에도 사용자가 같은 주파수 대역의 다른 채널을 시청하는데 아무런 제약을 두지 않게 구현하였다. 하나의 전송 스트림에는 소프트웨어 다운로드 데이터 채널뿐만 아니라 다른 일반 채널도 다중화되어 있기 때문에 소프트웨어 다운로드 채널을 선택하였다고 해서 같은 전송 스트림 내 다른 채널의 스트림들을 STB가 수신 못하는 것은 아니다. 소프트웨어를 다운로드하는 동안 느낄 수

있는 사용자의 불편함을 고려하여 다운로드가 소프트웨어 다운로드를 처리하는 동안에도 사용자가 다른 일반 채널을 시청할 수 있도록 구현하였다.

4. 다운로드의 컨트롤 메시지 처리 모듈

다운로더가 VCT를 처리하여 소프트웨어 다운로드 카루셀 스트림의 PID를 검출하였으면, 다운로드의 검출된 PID로 전송되는 카루셀 스트림으로부터 소프트웨어 다운로드 데이터를 추출하기 위해서 먼저 데이터 카루셀의 제어 메시지인 DSI 메시지를 필터링해야 한다. DSI 메시지를 필터링하기 위해서 다운로드의 PID인자에 VCT로부터 검출된 데이터 카루셀 스트림의 PID, table_id 인자에는 0x3B, transaction_id 인자에 0 또는 1을 설정한다. 소프트웨어 다운로드 규약에서 DSI 메시지는 Announcement DSI와 Signaling DSI로 구분된다. Announcement DSI는 미래에 데이터가 전송될 것이라는 것을 알려주기 위한 메시지이므로 다운로드의 실제 데이터가 전송될 시간까지 다운로드가 쓰는 자원을 해제하고 대기한다. 다운로드에서 Announcement DSI와 Signalling DSI의 구분은 scheduleDescriptor의 존재 여부를 체크하여 구분한다. 그전에 groupCompatibility Descriptor에서 제공하는 소프트웨어 다운로드의 대상이 되는 STB의 정보와 실제 STB의 정보가 일치할 경우에만 scheduleDescriptor의 존재 여부를 체크한다. groupCompatibility Descriptor는 전송되는 데이터가 어떤 종류의 STB들을 대상으로 하고 있는지를 기술하는 서술자이므로 다운로드의 groupCompatibility Descriptor에 기술된 정보가 수신장치의 정보와 일치할 때만 데이터를 다운로드 받는다. groupCompatibility Descriptor에서 제공되는 STB의 정보가 틀리면 다운로드를 수행할 필요가 없기 때문에 scheduleDescriptor의 존재 여부를 체크할 필요가 없다.

다운로더가 Announcement DSI 메시지를 수신하였을 경우에는 scheduleDescriptor의 startTime 값을 체크하여 그 시간까지 섹션 필터 자원을 해제하는 등 다운로드가 쓰는 자원을 해제한다. 다운로드의 루프를 돌면서 현재 시간이 startTime이 되었는지를 체크하여 그 시간이 되

있으면 실행 대기상태에서 벗어나 소프트웨어 다운로드 데이터 스트림을 필터링하기 위한 미들웨어 자원을 요청한다.

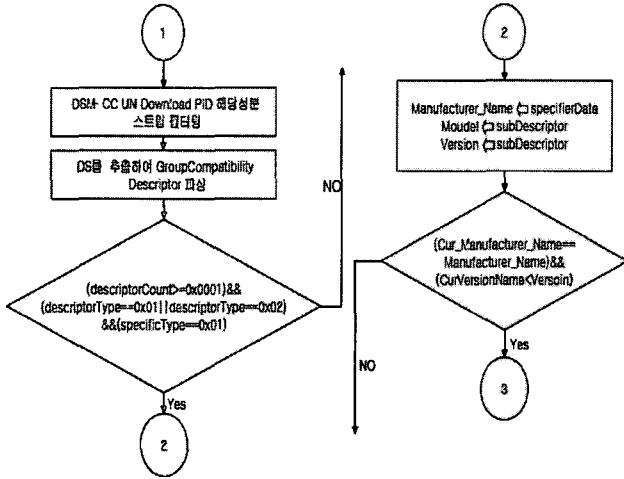


그림 3. DSI 메시지 섹션 필터링 및 처리
Fig. 3. DSI Message Section Filtering and Processing

다운로더는 현 STB 정보가 groupCompatibility descriptor에서 기술된 조건을 만족시키면 Announcement DSI와 Signaling DSI 메시지를 처리한다. 이 때, 다운로더는 DSI 메시지의 GroupInfoIndication에서 모듈들의 그룹 정보를 검출한다. GroupInfoIndication에서 제공하는 group_id는 해당 DII 메시지의 transaction_id이기 때문에 다운로더는 DII 메시지를 필터링하기 위해 검출된 그룹정보에서 group_id를 사용한다.

다운로더가 DII 메시지를 필터링하기 위해서 PID 인자는 데이터 카루셀 스트림의 PID, table_id 인자는 0x3B, table_id_extention 인자는 GroupInfoIndication에서 제공한 groupID로 설정된다. 다운로더가 각 그룹의 DII를 추출했으면, DII가 가리키는 일련의 모듈들을 추출한다. 이를 위해 module_id, module_size, version, length 등의 정보를 기술하는 moduleInfoByte를 추출한다. moduleInfoByte에서 제공하는 module_id와 module_size는 다운로더가 DDB 메시지를 추출하여 모듈 전체를 재구성하는데 중요한 정보들이다.

5. 다운로더의 DDB 필터링 및 소프트웨어 추출 모듈

다운로더가 각 module을 구성하는 DDB들을 필터링하기 위해서 PID 인자는 데이터 카루셀 스트림의 PID, table_id 인자는 0x3C, table_extention_id 인자는 DII 메시지의 moduleInfoByte에서 제공되는 module_id 값으로 설정된다. 이와 같은 조건으로 다운로더는 DDB 메시지를 필터링하여 실제 모듈 데이터를 재구성한다. 다운로더가 DDB 메시지에서부터 모듈들을 재구성하기 위해서는 같은 module id를 가지는 DDB 메시지의 block number에 따라 DDB 메시지들의 페이로드를 연결한다. DDB 메시지의 block number는 모듈내 해당 DDB 메시지의 위치를 나타내는 필드로서 모듈 데이터의 시작 부분을 담고 있는 DDB 메시지의 block number는 0 이고 이 후 순서에 따라 block number는 1씩 증가한다. 다음은 데이터 카루셀 스트림으로부터 실제 모듈 데이터를 추출하는 전체 과정을 보여주는 그림이다.

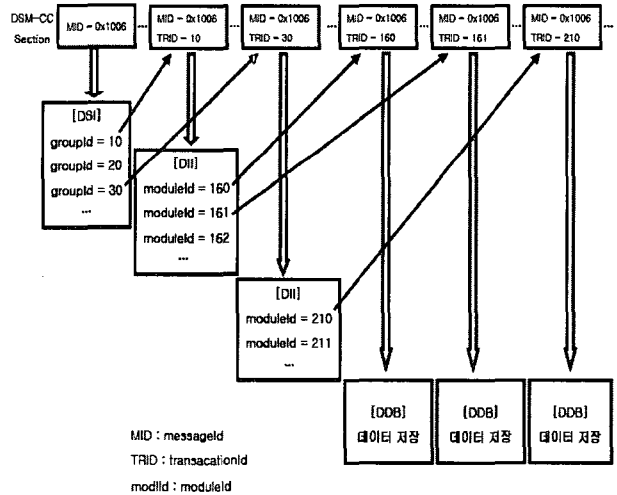


그림 4. DDB 메시지로 부터 모듈 데이터 추출^{[3][4]}
Fig. 4. Extracting module data from DDB messages

다운로더는 미리 지정된 임시 저장소에 추출된 소프트웨어를 저장한 후, 다운로더는 업데이트 로더를 호출한다. 다운로더에 의해 호출된 업데이트 로더는 임시 저장소에 저장된 새로운 소프트웨어를 STB에 설치하여 기존의 소프트웨어를 업데이트한다.

6. 업데이트 로더 구현

업데이트 로더는 전송 스트림에서 다운로드가 업데이트 할 소프트웨어를 추출하여 임시 저장소에 저장하면 기존의 소프트웨어를 임시 저장소의 새로 받은 소프트웨어로 업데이트시키는 프로그램이다. 업데이트 로더는 소프트웨어를 업데이트할 동안에 발생할 수 있는 예외상황에 대비하여 리커버러와 연동하여 소프트웨어의 업데이트를 완성한다. 업데이트 로더가 기존의 소프트웨어를 전송 스트림으로 부터 다운로드한 새로운 소프트웨어로 업데이트시키는 과정에서 사용자가 STB의 전원 케이블을 뽑는 등의 예외 사항이 발생할 수 있다. 이런 예외 상황에 대비한 처리가 업데이트 시작 전에 이루어져야 한다. 업데이트 로더는 이런 예외상황 발생에 대비하여 기존에 저장되어있는 소프트웨어를 복사하여 미리 지정된 소프트웨어 백업 장소에 저장한다. 그리고 복구용 로그파일에 업데이트가 시작됨을 기록한다. 복구용 로그파일에 기록되는 정보는 소프트웨어 업데이트가 성공적으로 완료되었는지 아님 실패했는지에 대한 플래그가 기록되어 있어 STB가 부팅할 때 이 로그 파일을 점검하여 오류 상황 플래그 값이 기록되어 있으면 STB 부팅시 리커버러를 자동 호출해 준다.

기존의 소프트웨어를 백업 장소에 복사해 두고 복구용 로그 파일에 업데이트 시작을 기록한 후에 업데이트 로더는 실제 소프트웨어를 업데이트한다. 업데이트가 성공적으로 완료되었으면 복구용 로그 파일에 이를 기록하고 업데이트용 로그 파일에 고유번호, 버전, 업데이트 완료 일시 등을 기록한다. 해당 로그 파일에 업데이트 정보를 기록한 후, 업데이트 로더는 실행 종료되고 다음 업데이트 때에 다시 다운로드에 의해 호출된다.

7. 리커버러 (Recoverer) 구현

리커버러는 STB가 부팅이 될 때, 업데이트 로더가 소프트웨어의 업데이트를 시행한 결과를 기록한 복구용 로그 파일 내용을 점검하여 업데이트가 실패했을 경우, 정상적인 STB의 구동을 보장하기 위하여 기존 시스템으로 복구를 수행하는 프로그램이다. 리커버러는 STB가 부팅이 될

때, `init script(/etc/init.d/rc.local)`를 통해 실행이 된다. 리커버러가 실행이 된 후, 복구용 로그 파일에 기록된 내용에 따라 업데이트 오류가 없었으면 리커버러는 바로 실행 종료가 되고 업데이트 오류가 있었다면 기존 시스템으로 복구하고 난 후 실행 종료 된다. 복구가 성공하면, 복구용 로그 파일에 성공여부를 기록하고 다음 STB 구동 시 리커버러가 재동작하지 않도록 한다.

복구용 로그 파일의 내용이 업데이트 실패일 경우, 리커버러의 소프트웨어 복구는 미리 지정된 해당 소프트웨어 백업 저장소에서 복사된 기존의 소프트웨어를 이용하여 수행이 된다. 복구가 성공하면, 복구용 로그 파일에 성공여부를 기록한다. 리커버러는 복구용 로그파일을 업데이트 로더와 공유한다.

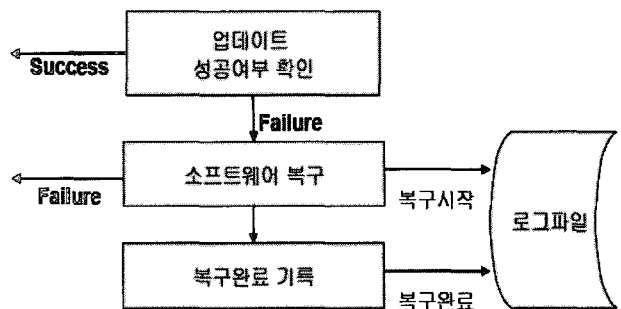


그림 5. 리커버러 동작 흐름도
Fig. 5. Flow Diagram of the Recoverer

8. ATSC와 DVB의 소프트웨어 업데이트 비교

본 논문의 소프트웨어 업데이트 시스템은 ATSC의 소프트웨어 다운로드 데이터 서비스를 지원할 수 있도록 구현되었다. 본 논문의 소프트웨어 업데이트 시스템과 같이 DVB 기반으로 소프트웨어 업데이트를 구현하기 위해서는 STB가 DVB에서 정의한 시스템 소프트웨어 업데이트 서비스를 처리할 수 있어야 한다. DVB에서 소프트웨어 업데이트를 위해 정의한 시스템 소프트웨어 업데이트 서비스는 ATSC와 같이 2계층 데이터 카루셀을 이용하여 전송된다.

VCT만 검색해도 소프트웨어 다운로드 데이터 서비스와 이를 전송하는 데이터 카루셀의 PID를 알 수 있는 ATSC

방식과 달리, DVB 방식에서는 소프트웨어 업데이트에 대한 정보 획득을 위해 최소 2단계 과정이 필요하다. 우선 NIT(Network Information Table)를 검색하여 방송 네트워크상에서 시스템 소프트웨어 업데이트 서비스가 전송되는 전송 스트림의 위치를 알아내고 해당 전송 스트림에서 PMT(Program Map Table)를 검색하여 시스템 소프트웨어 업데이트 서비스에 대한 데이터 카루셀의 PID를 알아 내는 최소 2단계 과정을 거쳐야 한다^[1].

DVB에서는 시스템 소프트웨어 업데이트 서비스에 대한 통지를 위해 단순 프로파일 소프트웨어 업데이트 서비스(simple profile software update services)와 UNT 확장 프로파일 소프트웨어 업데이트 서비스(Update Notification Table enhanced profile software update services)라는 두 가지 프로파일을 정의하고 있다^[1]. 단순 프로파일 소프트웨어 업데이트 서비스는 시스템 소프트웨어 업데이트 서비스에 대한 통지를 NIT와 PMT만으로 해결하는 프로파일이다. UNT 확장 프로파일 소프트웨어 업데이트 서비스는 시스템 소프트웨어 업데이트 서비스에 대한 스케줄 정보, 업데이트 대상 STB의 정보등과 같이 NIT와 PMT에서 제공될 수 없는 정보를 DVB SI 테이블인 UNT(Update Notification Table)를 통해 전송시켜주는 프로파일이다^[1]. ATSC에서는 업데이트의 스케줄 정보와 대상 STB 정보에 대한 처리는 실제 데이터 카루셀에 접근하여 데이터 카루셀에서 제공되는 디스크립터를 처리하는 과정에서 수행되어지는데 반해 DVB에서는 데이터 카루셀에 접근하기 전에 처리되는 UNT에서 그와 같은 정보를 처리한다. DVB에서는 ATSC의 VCT와 같이 전체널에 대한 모든 정보를 제공하는 테이블이 정의되어 있지 않으므로 여러 단계를 거쳐 소프트웨어 다운로드 데이터 카루셀에 접근한다. 반면에 DVB는 UNT를 통해 소프트웨어 데이터 카루셀에 접근하기 전에 시스템 소프트웨어 업데이트 서비스에 대한 충분한 사전 정보를 제공함으로써 소프트웨어 다운로드 데이터 카루셀에 대한 처리에 있어서는 ATSC에 비해 간단하다.

ATSC와는 달리 DVB는 UNT에서 제공되는 디스크립터에 특정 STB의 스마트카드 정보와 MAC 주소를 기술하여 특정 STB만 소프트웨어 업데이트를 수행할 수 있다^[1].

IV. 실험 방법 및 결과

개발 호스트에서 다운로더, 업데이트 로더, 리커버러를 개발하여 이 바이너리 파일을 인터넷을 통해서 SMARTV STB에 탑재시켜 테스트를 실행하였다. 실험은 전송 스트림을 2개의 가상 채널로 구성해 테스트를 하였다. 2개의 가상 채널은 비디오/오디오 스트림을 포함한 5개의 성분 스트림으로 구성된 가상 채널과 소프트웨어 다운로드 데이터 서비스로 구성된 가상 채널이다. ATSC 방식의 송출 시스템을 통해 2개의 가상 채널로 구성된 전송 스트림(TS) 파일을 SMARTV 셋탑 박스로 전송시켜 소프트웨어 업데이트 시스템을 검증하였다.

```

Service Location Descriptor
-----
SLD Tag : 0x01 Number Elements : 5
-----
Stream Info
Stream Type : 0x02
Elementary PID : 0x001
-----
Stream Info
Stream Type : 0x01
Elementary PID : 0x004
-----
Stream Info
Stream Type : 0x05
Elementary PID : 0x101
-----
Stream Info
Stream Type : 0x0b
Elementary PID : 0x102
-----
Stream Info
Stream Type : 0x01
Elementary PID : 0x104
-----
Program Number : 1
-----
Service Location Descriptor
-----
SLD Tag : 0x01 Number Elements : 1
-----
Stream Info
Stream Type : 0x01
Elementary PID : 0x004
-----
VCT End
  
```

그림 6. 다운로더의 VCT 검출 결과 화면

Fig. 6. Result for the Downloader to analyze a VCT

그림 6은 다운로더가 2개의 가상 채널 정보를 VCT에서 검출한 결과 화면이다. VCT의 Service Location Descriptor (SLD)를 해석하여 각 성분 스트림의 PID와 Stream Type

에 관한 정보를 보여주는 화면이다. service type이 0x05인 가상채널에서 stream type이 0x0B 인 PID는 결과화면에서 보듯 0x77로써 이 PID로 DSI, DII, DDB 메시지를 선택 필터링한 후 각 DSMCC 메시지를 처리하여 전송 스트림에서 업데이트 파일을 다운로드하였다. 그림 7은 각 DSMCC 메시지를 선택 필터링하여 처리한 결과화면이다.

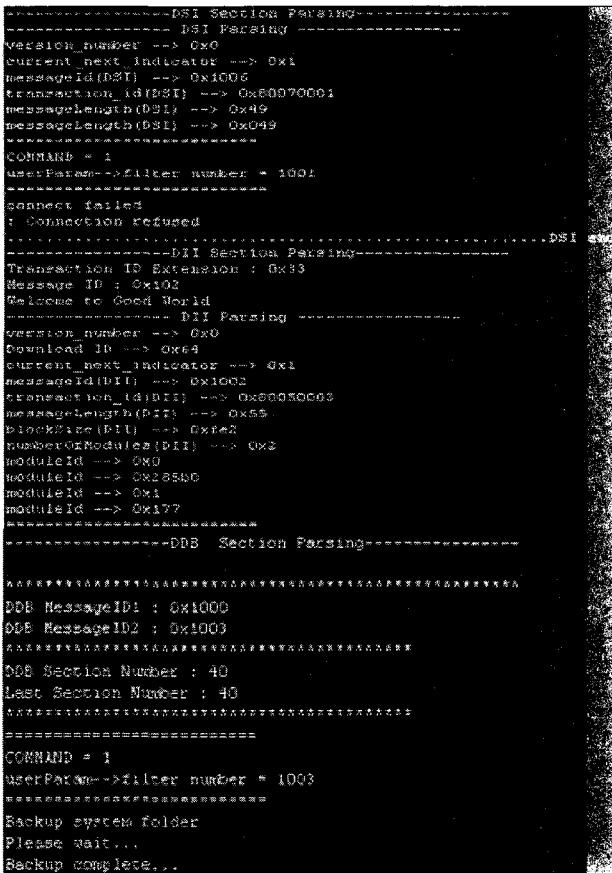


그림 7. DSI, DII & DDB 선택 필터링 및 처리 결과 화면
Fig. 7. Result of DSI, DII & DDB Section filtering

본 논문의 소프트웨어 업데이트 시스템은 소프트웨어를 다운로드하여 이를 업데이트하는 동안에도 사용자가 같은 주파수 대역의 다른 채널을 시청할 수 있도록 구현하였다. 그림 8은 소프트웨어 다운로드 데이터 서비스 처리 과정 중에도 일반 다른 채널을 시청하는 실험 결과 화면이다. 실험시, 일반 채널 시청중에 업데이트되고 있는 소프트웨

어는 SMARTV STB내에서 현재 시간을 얻어오는 실행 파일을 업데이트하였다. 이 실행 파일은 테스트용으로 만든 실행 파일로써 실험에서는 현재 시간을 분단위까지 얻어오는 실행 파일을 초단위까지 얻어오는 실행파일로 업데이트하도록 테스트하였다. SMARTV STB를 재부팅 한 후, 새로 다운로드된 실행파일로 실행됨을 확인하였다.

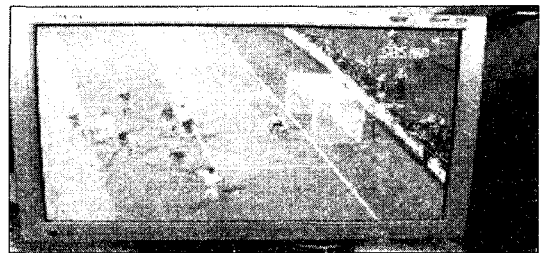


그림 8 소프트웨어 다운로드 데이터 서비스 처리 과정에서의 모니터 화면
Fig. 8. Monitor Screen while processing the Software Download Data Service

다운로더가 데이터 카루셀 스트림에서 업데이트되는 소프트웨어를 다운로드하면 업데이트 로더가 호출되는데 업데이트 로더는 업데이트 완료 후 업데이트 정보를 로그 파

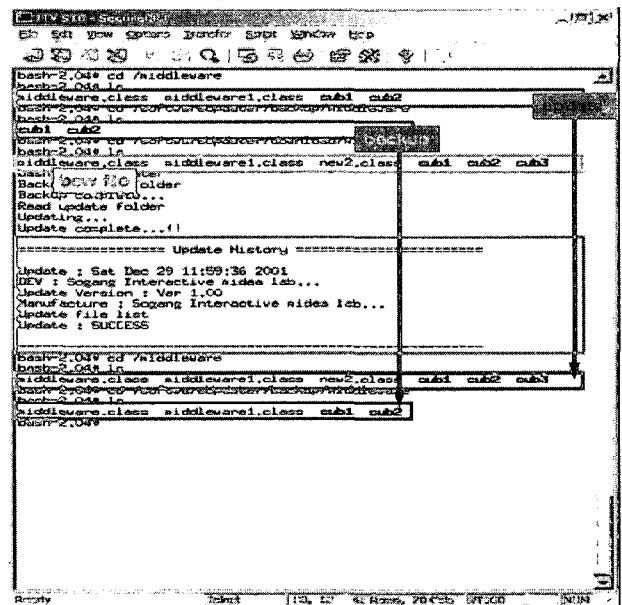


그림 9 업데이트 완료 후 로그 파일과 파일 목록
Fig. 9. Successfully updated file list and log file via Update Loader

일에 기록한다. 업데이트 로더는 기존의 파일을 백업하고 기존의 파일을 새로 받은 파일로 업데이트를 실시한다. 그림 9은 이런 과정과 업데이트 완료 후 로그 파일 내용과 파일 목록 변화를 보여준다.

본 논문에서 테스트한 ATSC 환경은 ATSC 전송 규격과 ATSC A/97 규격에 맞게 인코딩된 전송 스트림을 8-VSB 모듈레이터로 송출하여 이를 지상파 SMARTV STB에서 실험한 환경이다.

V. 결 론

최근 데이터 방송에 대한 관심이 고조되면서 혁신적이고 다양한 데이터 방송 콘텐츠에 대한 기대가 증가하고 있다. 이에 따라 실제 방송환경에서 다기능 고품질의 방송 콘텐츠를 서비스하기 위한 STB의 기능 지원이 필수적이다. 이와 관련하여 데이터 방송 환경에서 새로운 기술과 서비스가 등장할 때마다 이를 수용할 수 있도록 STB에 탑재된 기존의 소프트웨어를 업그레이드할 필요가 있다. 일반적으로 한번 가정 내에 보급된 STB의 소프트웨어 업그레이드가 용이하지 않다. 본 논문은 방송망을 통한 전송 스트림으로 STB의 업그레이드를 실현하도록 ATSC 소프트웨어 다운로드 데이터 서비스 규약을 수신기단에서 구현함으로써 디지털 방송 기술과 데이터 방송 기술에서 도입되는 신기술이나 새롭게 시도되는 서비스에 STB가 쉽게 적응할 수 있는 방송환경을 만들고자 하였다.

본 논문에서 구현하는 소프트웨어 업데이트 시스템은 소프트웨어를 포함하는 데이터 카루셀 스트림을 다운로드 받아 파싱하는 (1) 다운로드(Downloader), 추출된 소프트웨어를 설치하는 (2) 업데이트 로더(Update Loader), 그리고 예외상황이 발생하면 STB가 새로 부팅될 때, 로그 파일을 이용하여 소프트웨어를 이전 상태로 복구시키는 (3) 리커

버러(Recoverer)의 세 가지 부분으로 구성되어 있다. 본 논문의 소프트웨어 업데이트 시스템은 지상파 디지털 방송 규격인 ATSC에서 제시한 소프트웨어 다운로드 데이터 서비스 규약에 맞게 구현하고 ATSC 환경에서 이를 실험 검증하였다.

테스트 결과 미리 지정된 업데이트 폴더에 데이터 카루셀로 전송된 새로운 소프트웨어가 갱신됨을 확인할 수 있었다. 본 논문에서 구현된 다운로드가 탑재된 STB는 방송망에서 수신되는 플러그인 소프트웨어를 수용할 수 있게 됨으로써 다양한 멀티미디어 데이터를 처리할 수 있고 또한 미들웨어의 업그레이드를 통해 새롭게 바뀔 수 있는 데이터 방송 규격을 쉽게 수용할 수 있다.

참 고 문 헌

- [1] DVB. ETSI TS 102 006 Digital Video Broadcasting (DVB); Specification for System Software Update in DVB Systems 2002
- [2] ATSC. A/65B PROGRAM AND SYSTEM INFORMATION PROTOCOL FOR TERRESTRIAL BROADCAST AND CABLE 2003
- [3] ATSC. A/90 ATSC DATA BROADCAST STANDARD 2000
- [4] ATSC. A/91 Implementation Guidelines for the ATSC Data Broadcast Standard 2001
- [5] ATSC. A/97 Software Download Data Service 2004
- [6] Richard Chernock. "Data Broadcasting", McGraw-Hill, 2001
- [7] ISO/IEC 13818-6 General Coding of Moving Picture and Associated Audio: Digital Media Storage Media Command and Control 1996
- [8] ISO/IEC 13818-1 General Coding of Moving Picture and Associated Audio: System 1994
- [9] 정문열, 김용한, 백두원. 동기화된 데이터 방송을 위한 근사적인 NPT 재구성 기법. 2004년 한국방송공학회지, pp. 83~89
- [10] EYER. Program and System Information Protocol. McGraw-Hill, 2002
- [11] Interactive TV Web. <http://www.interactivetvweb.org>
- [12] Rick Whitaker. DTV Handbook 3rd edition, McGraw-Hill, 2000
- [13] Bin Yu, Klara Nahrstedt. A Real-time Software Solution For Resynchronizing Filtered MPEG2 Transport Stream, 2003

저 자 소 개



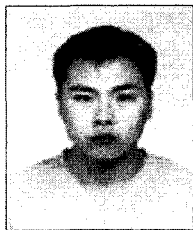
류 일 권

- 서강대학교 화학과 졸업 (BS)
- 서강대학교 영상대학원 미디어공학과 졸업 (MS)
- 현 에어코드 DTV 연구소 연구원
- 주관심분야 : STB, DTV



정 문 열

- 서울대학교 계산통계학과 졸업 (BS)
- 카이스트 전산학과 졸업 (MS)
- 일본 큐슈공과대학 정보공학부 조교수 역임
- 숭실대학교 컴퓨터학부 부교수 역임
- 현 서강대학교 영상대학원 미디어공학과 정교수



김 정 환

- 아주대학교 미디어학부 학사 (BS)
- 현 서강대학교 영상대학원 미디어공학과 석사과정
- 주관심분야 : iTV, IPTV, MPEG-4, 3DTV



최 진 수

- 경북대학교 전자공학과 학사 졸업 (BS)
- 경북대학교 전자공학과 석사 졸업 (MS)
- 경북대학교 전자공학과 공학 박사
- 한국전자통신연구원 데이터방송연구팀장 역임
- 현 TTA 데이터방송프로젝트그룹(PG312) 의장
- 주관심분야 : 멀티미디어 방송, 영상통신



방 건

- TTA 데이터방송프로젝트그룹(PG312) 간사
- 현 한국전자통신연구원 디지털방송연구단 선임연구원
- 주관심분야 : 데이터 방송 기술, 멀티미디어 방송, 영상통신